Serial No. 10/711,826
Group Art Unit 2168
Docket No: SVL920040022US1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

SUPPLEMENTAL APPEAL BRIEF – 37 C.F.R § 1.192

U.S. Patent Application 10/711,826 entitled:

"Method of Changing the Page Size of a DB2 Table Space While Keeping the Object Available"

Real Party in Interest: International Business Machines Corporation

Related Appeals and Interferences:

Applicants filed an Appeal Brief on 8/30/2007, which resulted in the Examiner reopening prosecution with a new non-final rejection which is the subject of this appeal.

Status of Claims:

Claims 1-22 are pending.

Claims 1-22 are rejected.

Claims 1, 3, 4 and 9 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. (U.S. Patent 6,965,899) in view of Teng et al. (U.S. Patent 6,460,048), and further in view of Bonner et al. (U.S. Patent 6,535,895) and further in view of the non-patent literature to Schiefer et al. entitled, "DB2 Universal Database Performance Turning," published June 1999.

Claims 2, 5-8 and 10-12 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Teng et al. and further in view of Bonner et al. and further in view of Schiefer et al. and further in view of Huras et al. (U.S. Publication 2001/0047360).

Claims 13, 14, 18 and 19 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Bonner et al. and further in view of Scheifer et al.

Claims 15-17 and 20-22 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Bonner et al. and further in view of Schiefer et al. and further in view of Teng et al.

Claims 1-22 are hereby appealed.

Status of Amendments:

No Amendments were presented after the Final Office Action of 02/23/2007.

Summary of Claimed Subject Matter:

(NOTE: All citations are made from the original specification, including the figures.)

The present invention, according to independent **claim 1**, provides for a method ***(see figure 2 in the application-as-filed)*** for updating object page size during reorganization of a table space in a database comprising the steps of: allocating a shadow data set for at least one object belonging to a first data set from said table space ***(see paragraph [0019] and step 200 of figure 2 in the application-as-filed)***; writing to a shadow control block corresponding to each of said allocated shadow data sets, a page size value larger than a page size value to be allocated; said larger page size value corresponding to said at least one object ***(see paragraph [0019] and step 204 of figure 2 in the application-as-filed)***; loading rows from said first data set of said table space into said allocated shadow data set; for each row loaded, reading each object corresponding to said loaded row from said table space and writing said read object to said allocated shadow data set ***(see paragraph [0019] and step 206 of figure 2 in the application-as-filed)***; and updating at least: said first data set of said table space with data from said shadow data set; a system catalog for said database with said larger page size value; and at least one database control block with said larger page size value; said at least one database control block corresponding to said first data set ***(see paragraph [0019] and step 210 of figure 2 in the application-as-filed)***.

The present invention, according to independent **claim 9**, provides for an article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements a method ***(see figure 2 in the application-as-filed)*** for updating object page size during reorganization of a table space in a database, said medium comprising modules implementing: allocating a shadow data set for at least one object belonging to a first data set from said table space ***(see paragraphs [0019], [0020], [0021], [0022] and step 200 of figure 2 in the application-as-filed)***; writing to a shadow control block corresponding to each of said allocated shadow data sets, a page size value larger than a page size value to be allocated; said larger page size value corresponding to said at least one object ***(see paragraphs [0019], [0020], [0021], [0022] and step 204 of figure 2 in the application-as-filed)***; loading rows from said first data set of said table space into said allocated shadow data set; for each row loaded, reading each object corresponding to said loaded row from said table space and writing said read object to said allocated shadow data set ***(see paragraphs [0019], [0020], [0021], [0022] and step 206 of figure 2 in the application-as-filed)***; and updating at least: said first data set of said table space with data from said shadow data set; a system catalog for said database with said larger page size value; and at least one database control block with said larger page size value; said at least one database control block corresponding to said first data set ***(see paragraphs [0019], [0020], [0021], [0022] and step 210 of figure 2 in the application-as-filed)***.

The present invention, according to independent **claim 13**, provides for a method for reorganizing a designated object of a database that has exceeded a current page size by: writing to a larger page, rows added to said designated object ***(see paragraphs [0009], [0010] and step 204 of figure 2 in the application-as-filed)***; permitting continual access to said designated object during said writing step ***(see paragraphs [0009] and [0010] in the application-as-filed)***; reading constituent rows from a plurality of existing pages corresponding to said designated object and subsequently copying said constituent rows to said larger page ***(see paragraphs [0009] and [0010] in the application-as-filed)***; and externalizing said designated object ***(see paragraphs [0009] and [0010] in the application-as-filed)***.

The present invention, according to independent **claim 18**, provides an article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements the reorganization of a designated object of a database that has exceeded a current page size; said medium comprising modules implementing: writing to a larger page, rows added to said designated object ***(see paragraphs [0009], [0010] and step 204 of figure 2 in the application-as-filed)***; permitting continual access to said designated object during said writing step ***(see paragraphs [0009] and [0010] in the application-as-filed)***; reading constituent rows from a plurality of existing pages corresponding to said designated object and subsequently copying said constituent rows to said larger page ***(see paragraphs [0009] and [0010] in the application-as-filed)***; and externalizing said designated object ***(see paragraphs [0009] and [0010] in the application-as-filed)***.

Grounds of Rejection to be Reviewed on Appeal:

1. Claims 1, 3, 4 and 9 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. (U.S. Patent 6,965,899) in view of Teng et al. (U.S. Patent 6,460,048), and further in view of Bonner et al. (U.S. Patent 6,535,895) and further in view of the non-patent literature to Schiefer et al. entitled, "DB2 Universal Database Performance Turning," published June 1999. Claims 2, 5-8 and 10-12 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Teng et al. and further in view of Bonner et al. and further in view of Schiefer et al. and further in view of Huras et al. (U.S. Publication 2001/0047360). Claims 13, 14, 18 and 19 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Bonner et al. and further in view of Scheifer et al. Claims 15-17 and 20-22 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Bonner et al. and further in view of Schiefer et al. and further in view of Teng et al. **With respect to pending claims 1-22, was a proper rejection made under 35 U.S.C. § 103(a) using existing USPTO guidelines**?

ARGUMENT:

1. With respect to pending claims 1-22, was a proper rejection made under 35 U.S.C. § 103(a) using existing USPTO guidelines?

Claims 1, 3, 4 and 9 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. (U.S. Patent 6,965,899) in view of Teng et al. (U.S. Patent 6,460,048), and further in view of Bonner et al. (U.S. Patent 6,535,895) and further in view of the non-patent literature to Schiefer et al. entitled, "DB2 Universal Database Performance Turning," published June 1999. Claims 2, 5-8 and 10-12 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Teng et al. and further in view of Bonner et al. and further in view of Schiefer et al. and further in view of Huras et al. (U.S. Publication 2001/0047360). Claims 13, 14, 18 and 19 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Bonner et al. and further in view of Scheifer et al. Claims 15-17 and 20-22 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Bonner et al. and further in view of Schiefer et al. and further in view of Teng et al.

Applicants contend that the above-mentioned specific combinations of references fail to provide many of the features of the Applicants' claims.

To establish a *prima facie* case of obviousness under U.S.C. §103, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. Additionally, the teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure (In re Vaeck, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991)).

Applicants contend that the above-mentioned specific combinations of references fail to provide many of the features of the Applicants' claims.

With respect to independent claims 1 and 9, the Examiner contends that the Subramaniam and Teng references, in combination, teach all the features of Applicants' claims 1 and 9. Applicants respectfully disagree with this assertion.

Applicants' claim 1 provides for a method for updating object page size during reorganization of a table space in a database comprising the steps of: allocating a **shadow data set** for at least one object belonging to a first data set from said table space; writing to a **shadow control block** corresponding to each of said allocated shadow data sets, **a page size value larger than a page size value to be allocated**; said larger page size value corresponding to said at least one object; loading rows from said first data set of said table space into said allocated shadow data set; for each row loaded, reading each object corresponding to said loaded row from said table space and writing said read object to said allocated shadow data set; and updating **at least**: said first data set of said table space with data from said shadow data set; a system catalog for said database with said larger page size value; **and** at least one database control block with said larger page size value; said at least one database control block corresponding to said first data set.

Applicants claim 9 provides for an article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements the method as substantially described in claim 1.

Subramaniam et al., by contrast, teaches a method for modifying a target table within a relational database. Subramaniam's method comprises the steps of: creating a revised table that has one or more attributes that are different than corresponding attributes of the target table. While the revised table is being created, users are prevented from accessing the revised table, but are allowed to access the target table. According to Subramaniam's method, when creation of the revised table is complete, the target table is locked and the revised table is synchronized with the target table, causing all subsequent attempts to access the target table to access the revised table.

Teng et al., also by contrast, teaches a method for reorganizing a database object, wherein the database object is comprised of at least one database file. Teng's method comprises the steps of: providing source database files including data for the database objects subject to a reorganization, wherein the source database files have source names; creating shadow copies of the source database files; generating shadow names for the shadow copies, wherein the source names and corresponding shadow names are different; and reorganizing data in the shadow copies including database objects, wherein after the reorganization, the shadow names are used to access the database files for the reorganized database objects.

Bonner et al., also by contrast, teaches a technique to avoid processing well clustered LOBs during reorganization of a LOB table space. Bonner's method involves inserting or updating data into a portion of the table space, setting a flag in a space map which is associated with the data to indicate whether the data was well inserted and, when reorganizing the table space, avoiding reorganization of the portion of the table space in which the data was well inserted.

Schiefer et al., also by contrast, as the title suggests, teaches auto-configuration tools for DB2 universal performance tuning.

In withdrawing the Final Office Action of 02/23/2007 and in responding to the Appeal Brief of 11/28/2007, the Examiner agrees with the arguments presented in the Appeal Brief of 11/28/2007 regarding the lack of showing in both Subramanian and Teng for the feature of a page size value **larger than** a page size value to be allocated. However, the Examiner, in the most recent response of 11/28/2007, appears to have erroneously concluded that such a feature is shown in the newly cited Bonner and Schiefer references.

Specifically, in the "Response to Arguments" section on page 16 of the Office Action of 11/28/2007, the Examiner states that Bonner in column 6, lines 20-24 discloses a maximum table size as defined by the page size. Column 6, lines 20-24 of Bonner is reproduced verbatim below:

> "When inserting a LOB, one or more LPB low-level space map pages 302, 304 are accessed to find LOB pages that may be allocated to the LOB. *All LOB pages within a single table space have the same page size. A page may contain, for example, 4096 bytes.*" (emphasis added).

As can be seen from the Examiner's citation above, Bonner merely mentions that **all large object (LOB) can be pages within a single table space having the same page size, wherein a page may contain 4096 bytes**. The above citation merely references the **allocation of a page size of a known size - for example, a page size of 4096 bytes**.

However, Applicants wish to note that Applicants pending claims 1 and 9 **do not just make a mention of a page size**, but specifically mention the step of "writing to a shadow control block .... **a page size value *larger than a page size to be allocated***". Applicants respectfully assert that **the Examiner's citation merely teaches the allocation of a page size of 4096, but fails to teach or suggest increasing the page size value, once it is allocated**.

For further support to Applicants' arguments, Applicants respectfully request the Board of Patent Appeals and Interferences to review column 6, lines 12-15, which says that "Each LOB page is allocated to one LOB, even if the LOB uses a portion of the LOB page. For example, one LOB may be stored on 17 and a half LOB pages, but the LOB page that is half used is not allocated to any other LOB". This citation reinforces Applicants' contention that the Bonner reference merely teaches allocation of pages but makes NO mention or suggestion for allocating a page size value that is larger than a page size value to be allocated. Absent such a teaching, Applicants respectfully submit that the Bonner reference in combination with the Subramaniam, Teng, and Schiefer CANNOT teach all the features of Applicants' pending claims 1 and 9.

The Examiner's other citation in the "Response to Arguments" section, the Examiner cites page 18, first paragraph of Schiefer as teaching the feature of increasing page size. Paragraph 1 on page 18 of Schiefer is provided verbatim below:

> "We begin with the physical database design. The use of multiple table spaces can give fine-grained control over the placement of data. For example, separating data and indexes into different table spaces or placing table data on only a subset of the disks available. The use of multiple table spaces also allows the database to be managed at a more granular level since many of the database utilities operate at a concurrency granularity of a table space (e.g. load, backup/restore). *The characteristics of each table space can also be controlled, including defining a page size ranging from 4 KB to 32 KB, the size of an extent placed on each disk, the number of pages to prefetch from the table space, and a summary of the performance characteristics of the underlying disk devices*." (emphasis added).

As can be seen from the Examiner's citation above, Schiefer merely mentions that **a page size can be defined in the range of 4 KB to 32 KB**. As with the Bonner reference, the above citation merely references the **allocation of a page size of a known size - for example, a page size in the range of 4KB to 32KB**. As stated earlier, Applicants wish to note that Applicants' pending claims 1 and 9 **do not just make a mention of a page size**, but specifically mention the step of "writing to a shadow control block .... **a page size value *larger than a page size to be allocated***". For example, if a page size to be allocated is 4 KB, the present invention teaches the feature of allocating a page size that is larger than the allocated 4KB. Applicants respectfully assert that **the Examiner's citation merely teaches the allocation of a page size, but fails to teach or suggest increasing the page size value, once it is allocated**. Absent such a teaching, Applicants respectfully submit that the Schiefer reference in combination with the Subramaniam, Teng, and Bonner CANNOT teach all the features of Applicants' pending claims 1 and 9.

Further, Applicants argue that even for argument purposes if the teachings of Bonner and Schiefer were combined with the teachings of Subramaniam and Teng, it would merely provide for allocation of page size of predetermined value and would NOT provide for **writing to a shadow control block a page size value** ***larger than a page size value to be allocated.***

Applicants agree with the Examiner's conclusion on page 4 of the latest Office Action of 11/28/2007 that Subramaniam reference does NOT teach claim 1 and 9's feature of "updating **at least**: said ***first data set*** of said table space with data from said shadow data set; a ***system catalog*** for said database with said larger page size value; ***and*** at least one ***database control block*** with said larger page size value; said at least one database control block corresponding to said first data set". However, Applicants respectfully maintain that such a feature is not remedied by the Teng reference.

Specifically, for support, the Examiner cites column 6, lines 55-65 of Teng as teaching this feature of claims 1 and 9. Column 6, lines 55-65 of Teng merely reference Figure 2 which teaches the reorganization of "shadow copies of data sets". However, Applicants respectfully assert that neither Teng's Figure 2 nor Teng's description of Figure 2 attempts to teach or suggest **updating a system catalog for said database with said larger page size value**". Applicants also assert that Teng's reorganization procedure shown in Figure 2 and accompanying description fail to teach or suggest **updating one database control block with said larger page size value**". Further, absent in the Examiner's citations and the entire Teng reference is a teaching for **updating a first data set of the table space with data from a shadow set**. As **each** of the above-mentioned features are absent in Subramaniam and Teng references, it would be moot to argue that a **combined teaching for all three updates** can be found in the Subramaniam and Teng references (see, for example, claim 1 – "updating **at least**: said first data set of said table space with data from said shadow data set; a system catalog for said database with said larger page size value; **and** at least one database control block with said larger page size value"). Absent such a teaching, Applicants respectfully submit that the combination of Subramaniam, Teng, Schiefer and Bonner CANNOT teach all the features of Applicants' pending claims 1 and 9.

Applicants wish to note that the above-mentioned arguments for independent claims 1 and 9 substantially apply to dependent claims 2-8 and 10-12 as they inherit all the features of the claim from which they depend.

Claims 13, 14, 18 and 19 are rejected under 35 U.S.C. §103(a) as being unpatentable over Subramaniam et al. in view of Bonner et al. and further in view of Scheifer et al.

Claim 13 provides for a method for reorganizing a **designated object of a database that has exceeded a current page size** by: writing to a **larger page**, rows added to said designated object; permitting continual access to said designated object during said writing step; reading constituent rows from a plurality of existing pages corresponding to said designated object and subsequently copying said constituent rows to said **larger page**; and externalizing said designated object. Applicants' claim 18 provides for an article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements the method as substantially described in claim 13.

As with independent claims 1 and 9, the Examiner in the recent rejection of 11/28/2007, states that Bonner and Scheifer teach the "larger page" features of independent claims 13 and 18. However, as was clearly explained above, the new citations by the Examiner merely teach the **allocation of page size of predetermined value** and would NOT provide for **writing to a** ***larger page size (once the current page size has been exceeded) rows added to a designated object***.

Absent such a teaching, Applicants respectfully submit that the combination of Subramaniam, Bonner, and Schiefer CANNOT teach all the features of Applicants' pending claims 13 and 18.

Applicants wish to note that the above-mentioned arguments for independent claims 13 and 18 substantially apply to dependent claims 14-17 and 19-22 as they inherit all the features of the claim from which they depend.

Applicants also respectfully requests the Board of Patent Appeals and Interferences to also consider fully the arguments presented in the Appeal Brief submitted on 8/30/2007.

Hence, at least for the reasons set forth above and the reasons set forth in the previous Appeal Brief of 8/30/2007, Applicants respectfully maintain that the above-mentioned specific combinations of references fail to provide many of the features of Applicants' pending claims 1-22. Hence, Applicants respectfully assert that the Examiner has failed to establish a *prima facie* case of obviousness, and further assert that an improper 35 U.S.C. §103(a) rejection was issued with regards to claims 1-22.

SUMMARY

As has been detailed above, none of the references, cited or applied, provide for the specific claimed details of applicant's presently claimed invention, nor render them obvious. It is believed that this case is in condition for allowance and reconsideration thereof and early issuance is respectfully requested.

As this Appeal Brief has been timely filed within the set period of response, no fee for extension of time is required. However, the Commissioner is hereby authorized to charge any deficiencies in the fees provided, including extension of time, to Deposit Account No. 09-0460.

Respectfully submitted by
Applicant's Representative,

/ramraj soundararajan/

Ramraj Soundararajan
Reg. No. 53,832

IP AUTHORITY, LLC
4821A Eisenhower Ave
Alexandria, VA 22304

February 28, 2008

Claims Appendix:

1. **(Previously Presented)** A method for updating object page size during reorganization of a table space in a database comprising the steps of:

(a) allocating a shadow data set for at least one object belonging to a first data set from said table space;

(b) writing to a shadow control block corresponding to each of said allocated shadow data sets, a page size value larger than a page size value to be allocated; said larger page size value corresponding to said at least one object;

(c) loading rows from said first data set of said table space into said allocated shadow data set; for each row loaded, reading each object corresponding to said loaded row from said table space and writing said read object to said allocated shadow data set; and

(d) updating at least: said first data set of said table space with data from said shadow data set; a system catalog for said database with said larger page size value; and at least one database control block with said larger page size value; said at least one database control block corresponding to said first data set.

2. (**Original**) A method for updating object page size during reorganizing a table space in a database, as per claim 1, wherein said method further comprising the steps of:

prior to said shadow data set allocation, blocking write access to said first data set from said table space; and

subsequent to said updating said table space, said database system catalog, and said at least one database control block, allowing write operations related to said first data set to proceed.

3. **(Previously Presented)** A method for updating object page size during reorganizing a table space in a database, as per claim 1, wherein said method is implemented across network elements.

4. **(Previously Presented)** A method for updating object page size during reorganizing a table space in a database, as per claim 3, wherein said across network elements is any of the following: local area network (LAN), wide area network (WAN), or the Internet.

5. **(Original)** A method for updating object page size during reorganizing a table space in a database, as per claim 1, wherein said loading is further comprised of:

(a) concurrently loading rows corresponding to said at least one object from said table space into said shadow data set and extracting index keys for each loaded row; said shadow data set allocated for each of said at least one object and associated indices, and

(i) for each of said loaded rows, identifying columns representing data corresponding to said at least one object; and

(ii) for each column representing data corresponding to said at least one object, reading data from said table space; said data read using row information from a currently loaded row; and writing said data corresponding to said at least one object to said shadow data set.

6. (Original) A method for updating object page size during reorganizing a table space in a database, as per claim 5, wherein said method further comprising the steps of:

prior to said concurrent loading of rows and extracting of index keys, unloading rows from said table space; and

sorting said unloaded rows; said sorted rows subsequently loaded into said shadow data set in said loading step.

7. (Original) A method for updating object page size during reorganizing a table space in a database, as per claim 5, wherein said method is implemented across network elements.

8. (Original) A method for updating object page size during reorganizing a table space in a database, as per claim 7, wherein said across network elements is any of the following: local area network (LAN), wide area network (WAN), or the Internet.

9. (Original) An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements a method for updating object page size during reorganization of a table space in a database, said medium comprising modules implementing:

(a) allocating a shadow data set for at least one object belonging to a first data set from said table space;

(b) writing to a shadow control block corresponding to each of said allocated shadow data sets, a page size value larger than a page size value to be allocated; said larger page size value corresponding to said at least one object;

(c) loading rows from said first data set of said table space into said allocated shadow data set; for each row loaded, reading each object corresponding to said loaded row from said table space and writing said read object to said allocated shadow data set; and

(d) updating at least: said first data set of said table space with data from said shadow data set; a system catalog for said database with said larger page size value; and at least one database control block with said larger page size value; said at least one database control block corresponding to said first data set.

10. (Original) An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements a method for updating object page size during reorganization of a table space in a database, as per claim 9, wherein:

prior to said shadow data set allocation, blocking write access to said first data set from said table space; and

subsequent to said updating said table space, said database system catalog, and said at least one database control block, allowing write operations related to said first data set to proceed.

11. (Original) An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements a method for updating object page size during reorganization of a table space in a database, as per claim 9, wherein said loading is further comprised of:

(a) concurrently loading rows corresponding to said at least one object from said table space into said shadow data set and extracting index keys for each loaded row; said shadow data set allocated for each of said at least one object and associated indices, and

(i) for each of said loaded rows, identifying columns representing data corresponding to said at least one object; and

(ii) for each column representing data corresponding to said at least one object, reading data from said table space; said data read using row information from a currently loaded row; and writing said data corresponding to said at least one object to said shadow data set.

12. (Original) An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements a method for updating object page size during reorganization of a table space in a database, as per claim . A method for updating object page size during reorganizing a table space in a database, as per claim 11, wherein said method further comprising the steps of:

prior to said concurrent loading of rows and extracting of index keys, unloading rows from said table space; and

sorting said unloaded rows; said sorted rows subsequently loaded into said shadow data set in said loading step.

13. (Previously Presented) Reorganizing a designated object of a database that has exceeded a current page size by:

(a) writing to a larger page, rows added to said designated object;

(b) permitting continual access to said designated object during said writing step;

(c) reading constituent rows from a plurality of existing pages corresponding to said designated object and subsequently copying said constituent rows to said larger page; and

(d) externalizing said designated object.

14. (Previously Presented) Reorganizing a designated object of a database that has exceeded a current page size, as per claim 13, wherein during said copying, constituent rows of said designated object are re-arranged in physical storage to eliminate fragmentation.

15. (Previously Presented) Reorganizing a designated object of a database that has exceeded a current page size, as per claim 13, wherein said database is comprised of: a plurality of index values and a system catalog.

16. (Previously Presented) Reorganizing a designated object of a database that has exceeded a current page size, as per claim 15, wherein during said copying, data in said constituent rows is compacted and is stored, on contiguous pages in physical storage, in accordance with one of said plurality of index values.

17. (Previously Presented) Reorganizing a designated object of a database that has exceeded a current page size, as per claim 15, wherein control information associated with said system catalog is updated to reflect a change in page size corresponding to said externalized designated object.

18. (Previously Presented) An article of manufacture comprising a computer usable medium having computer readable program code embodied therein which implements the reorganization of a designated object of a database that has exceeded a current page size; said medium comprising modules implementing:

a) writing to a larger page, rows added to said designated object;

b) permitting continual access to said designated object during said writing step;

c) reading constituent rows from a plurality of existing pages corresponding to said designated object and subsequently copying said constituent rows to said larger page; and

d) externalizing said designated object.

19. (Previously Presented) An article of manufacture comprising a computer usable medium, as per claim 18, wherein during said copying, constituent rows of said designated object are re-arranged in physical storage to eliminate fragmentation.

20. (Previously Presented) An article of manufacture comprising a computer usable medium, as per claim 18, wherein said database is comprised of: a plurality of index values and a system catalog.

21. (Previously Presented) An article of manufacture comprising a computer usable medium, as per claim 20, wherein during said copying, data in said constituent rows is compacted and is stored, on contiguous pages in physical storage, in accordance with one of said plurality of index values.

22. (Previously Presented) An article of manufacture comprising a computer usable medium, as per claim 20, wherein control information associated with said system catalog is updated to reflect a change in page size corresponding to said externalized designated object.

Evidence Appendix

None

Related Proceedings Appendix

None